Centralized Random Backoff for Collision Resolution in Wi-Fi Networks

Centralized Random Backoff for Collision Resolution in Wi-Fi Networks

Jinho D. Kim, David I. Laurenson, and John S. Thompson

Abstract

Wi-Fi devices operate following the 802.11 DCF (Distributed Coordination Function) in order to fairly use the channel that the devices share. However, the throughput performance of the Wi-Fi networks is known to be degraded due to packet collisions. So, we propose a novel multiple access protocol, called centralized random backoff (CRB) for collision free Wi-Fi networks. In CRB, after a successful reception of a data frame from a station, the AP allocates a unique backoff state to the station by means of the ACK frame. We evaluate its performance by comparing to that of a deterministic backoff mechanism. Evaluation results show that CRB significantly improves the throughput performance by reducing collisions, and it allows a larger number of nodes to operate in a collision free state without dynamic parameter adjustment.

Index Terms

Medium access control, MAC protocol, random access, distributed access, distributed coordination function, random backoff, collision resolution, and fairness.

I. Introduction

Wi-Fi technology following the IEEE 802.11 standard has become popular through operating with a simple decentralised MAC (Medium Access Control) protocol in unlicensed radio bands. Generally, Wi-Fi devices use the 802.11 DCF protocol to schedule their transmissions. The DCF implements random access by exponentially increasing the contention window size for each transmission failure in order to avoid consecutive collisions. Because each station independently selects a random number as its backoff count before transmitting, packet collisions can occur when two or more stations contend to transmit simultaneously.

The authors are with the Institute for Digital Communications (IDCOM) in School of Engineering, the University of Edinburgh, EH9 3JL, U.K. (e-mail: j.kim@ed.ac.uk; dave.laurenson@ed.ac.uk; john.thompson@ed.ac.uk).

Collision (or contention) resolution is one of the key goals of wireless MAC protocols [1]. In the 802.11 DCF, given a number of active nodes (=n) the collision probability (=p) tends to decrease as the minimum (or the initial) contention window size (=W_0) increases. However, the channel idle time (i.e. empty time slots) also increases with the value W_0. The fact that an optimum value of W_0 exists for a given value n was explained with a Markov chain model in [2]. The chain model was further developed for more practical conditions (such as a finite retry limit, imperfect channels, and unsaturated traffic) [3]–[6], and several ideas were proposed to dynamically estimate the value n for timely adjustment of the optimum value of W_0 [7], [8]. Furthermore, in order to improve the throughput performance (while maintaining fairness) various mechanisms for tuning the contention window sizes were proposed in [9]–[13]. Recently the 802.11ax project [14] has been tackling the challenging goal of improving the throughput in dense user environments. However, the fact that collisions increase with n is still a feature of such systems, causing throughput degradation. In addition, when the value n varies over time, fast adaptation of the optimum value of W_0 is still a complex issue in practice.

A *collision free* Wi-Fi network, where the value p is zero, has been studied in [15]–[23]. In Early Backoff Announcement (EBA) [15], a station announces its future backoff count using the MAC header of its transmitted data frame. All the neighbouring stations that receive the backoff count avoid collisions by excluding the same backoff count when selecting their future backoff value. However, the performance of EBA is significantly limited in practice, because some of the neighbouring stations may not be able to overhear the announced backoff count in the data frame. This is because different data rates have a different transmission coverage to each other[1].

According to [17]–[20], a collision free Wi-Fi network can be achieved by each active node setting its backoff counter to a deterministic value upon a successful packet transmission. This deterministic backoff mechanism is called CSMA/ECA[2] in [19] and also called semi-random backoff (SRB) in [20]. In the case of a failed packet transmission, the station reverts to the standard random backoff procedure of DCF. However, the maximum value n that can operate in a collision free state (=n_{max}) is limited to the value $\frac{W_0}{2}$. So, to support a larger number of nodes in a collision free state, the value W_0 has to be increased. However, channel idle time also tends to increase with the value W_0. Because of this, when the value n is assumed to change

[1]e.g. the coverage of a station using a 54 Mbps data rate is much smaller than that of a station using a 6 Mbps data rate.

[2]Carrier Sense Multiple Access with Enhanced Collision Avoidance

over time, the deterministic backoff requires dynamic adjustment of the optimum value of W_0. However, timely adjustment of the value W_0 is still a complex issue[3].

We propose a novel collision resolution solution called *centralized random backoff (CRB)*, in which backoff information is generated by the AP and allocated to the connected stations by means of ACK frames. The ACK frames from the AP are generally more reliable than the data frames from distributed stations. Like the deterministic backoff, CRB achieves a collision free state after a given convergence time[4](=T_{cvg}). However, it is expected to be a more effective solution than the deterministic backoff, because the value n_{max} in CRB is not limited to the value $\frac{W_0}{2}$, but increases with the convergence time[5]. This means that given sufficient convergence time a larger number of nodes can (automatically) operate in a collision free state without dynamic parameter adjustment.

CRB is different to the 802.11 PCF (Point Coordination Function). It is known that the PCF has several limitations. First, traffic to be sent under DCF during a CP (Contention Period) must wait until the end of the CFP (Contention Free Period) before channel access can be gained. This can severely impact delay sensitive traffic. Second, optimization of the ratio between CFP and CP may be too slow to cope with the variation of n in time. However, in CRB each time slot is randomly reserved by the virtual backoff algorithm (VBA) running in the AP. In order to select a time slot to be reserved, VBA imitates the standard DCF protocol[6]. This results in randomly distributed empty time slots over time (like empty time slots in the DCF), which is necessary to support new entrants using delay sensitive applications.

We present a Markov chain that models CRB, and analyse its performance. It is shown by analysis and simulation that CRB achieves a collision free state after a convergence time, and the throughput performance has been improved without a dynamic parameter adjustment. While

[3]CSMA/ECA [19] proposed a centralized (and explicit) adjustment using beacon frames, and SRB [20] suggested a distributed (and implicit) adjustment. Due to the beacon interval, the first may not be fast for timely adjustment. The second may not be fair when the value n varies over time, because one of the stations may use a different parameter with that of the other stations.

[4]i.e. the time period required for the wireless network to move toward a collision free state, during which the network automatically moves from the distributed mode to the centralized mode.

[5]Theoretically, the value n_{max} in CRB is limited to the maximum contention window size (=W_m). According to the standard, the default values of W_0 and W_m are 16 and 1024 respectively.

[6]Legacy Wi-Fi devices perform random access following the CSMA/CA (Carrier Sense Multiple Access with Collision Avoidance) to transmit signals. This means that a deterministic TDMA (Time Division Multiple Access) scheduling would not compatible with the Wi-Fi networks due to interferences with the signals transmitted by the legacy devices. Specifically, a deterministic TDMA scheduling would cause a fairness issue between the stations using a deterministic TDMA and the legacy stations; otherwise, it will require a dynamic parameter adjustment to the distributed stations to maintain fairness. However, the dynamic parameter adjustment is still a very complex issue when the number of users varies over time.

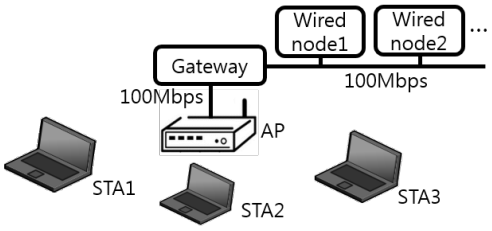


Fig. 1. Wi-Fi network configuration.

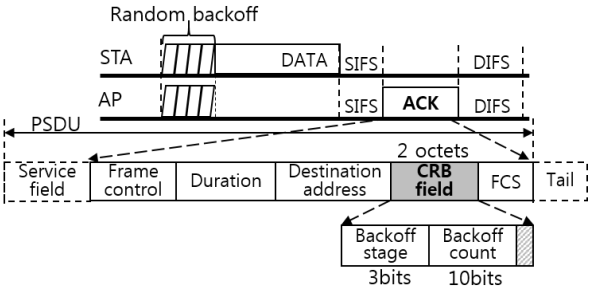


Fig. 2. CRB field.

the analysis results of collision free states closely match with the simulation results, there exists a small gap between the analysis results and simulation results in the collision prone states[7]. This is because the Markov chain model implicitly includes simplifications when the network states varies during the convergence time period[8].

The rest of the paper is organized as follows. In Section II, we review the system model to be discussed. In Section III, we explain how CRB operates in detail. In Section IV, we theoretically analyse the performance of CRB. In Section V, we present simulation results for validating the analysis model. Section VI concludes this paper.

II. System Model

As shown in Fig. 1, we consider a Wi-Fi network model operating in the infrastructure mode in the IEEE 802.11 standard. A wireless network consisting of a single AP together with all the connected stations (STAs) is called a BSS (Basic Service Set). We assume that the AP and the connected STAs transmit signals with a constant transmission power (=16 dBm). All the nodes support only the 802.11a physical layer (PHY), and they remain stationary (i.e. no mobility). The log distance propagation loss model (exponent=3) in [24], which predicts the received signal power as a deterministic function of distance, is assumed to apply to the transmitted signals.

In order to evaluate the throughput performance when using CRB in saturation conditions[9], we use UDP (User Datagram Protocol) data packets in simulation and assume that a real time streaming application is being used by each user. In addition, we also perform an initial

[7]Collision prone states means the network states during the convergence time period, where the value p is larger than zero.

[8]As seen in Section IV-B, we explain that a Markov chain modelling network states during the convergence time period is too complex to find a closed form solution.

[9]i.e. where the transmission queue of each node is assumed to be always non-empty and each active node immediately attempts to transmit a packet after the completion of each transmission. The saturation assumption means that the queueing dynamics are negligible.

assessment of CRB performance in unsaturated conditions. For numerical analysis in Section IV we assume interference from adjacent wireless networks is negligible, and focus on a single BSS environment. This assumption enables a tractable numerical analysis in this paper. To demonstrate practicality, in Section V we evaluate the proposed protocol in a number of situations such as overlapping APs, mixed nodes, and hidden nodes.

III. Centralized Random Backoff Protocol

In the 802.11 DCF, each node independently selects a random number as its backoff count before transmitting. This means when two or more nodes contend to transmit simultaneously, some of the nodes can have the same backoff count, causing packet collisions. However, in CRB, the AP internally generates a unique backoff state and allocates it to each node. When the AP has received a successful data frame from a source node and discovered that the node has more data packets to send, the AP allocates a backoff state to the node using the ACK frame shown in Fig. 2, where two octets are added to carry the backoff state. (We assume that one bit of the *More Data* field in the MAC header of the data frame can be used to inform the AP that the source node has more data packets to send.)

The backoff state includes two numbers: a backoff stage (BS) and a backoff count (BC). These are generated by the AP after successfully receiving the data frame from the source node and before transmitting the ACK frame. When the source node has successfully received the ACK frame with the backoff state and uses the backoff state for transmitting the next data frame, we call it a *synchronized CRB node (SCN)* and the allocated backoff count is called a *synchronized backoff count (SBC)*.

If the transmission of the data frame is unsuccessful and the source node fails to receive the ACK frame, then in order to start contention again for retransmission the source node generates a new backoff state by itself like a station operating in the current IEEE 802.11 DCF. In this case, we call the source node an *unsynchronized CRB node (UCN)* and the (independently) generated backoff count is called an *unsynchronized backoff count (UBC)*.

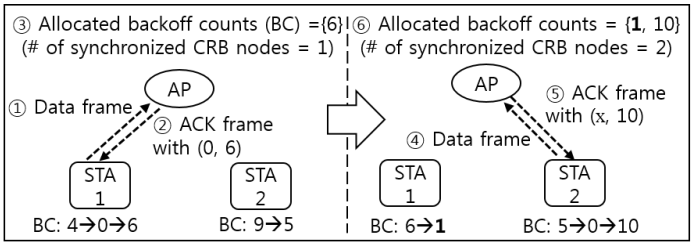
The value of BS (=i) is an integer in the range $[0, m]$, where m represents the maximum value of BS. The value of BC (=k) is an integer in the range $[0, W_i - 1]$, where W_i represents the contention window size at BS i. The value of W_i is $2^i W_0$, where W_0 represents the minimum contention window. In this paper, we assume that $m = 6$ and $W_0 = 16$ by default. Therefore, the value of W_m is 1024.

After successfully receiving the data frame from the source node and before transmitting the ACK frame to the source node, the AP internally generates a backoff state (which is to be included in the ACK frame) as described in Fig. 3. In Fig. 3a, after successfully receiving a data frame from STA1 (i.e. after step ① and before step ②), the AP uniformly selects a random number from the range [0, 15] *(virtual backoff stage* 0*)* as a backoff count to be included in the ACK frame (e.g. number 6 as shown in Fig. 3a). In this case, the value of backoff stage to be included in the ACK frame is zero (as seen in Fig. 3b).

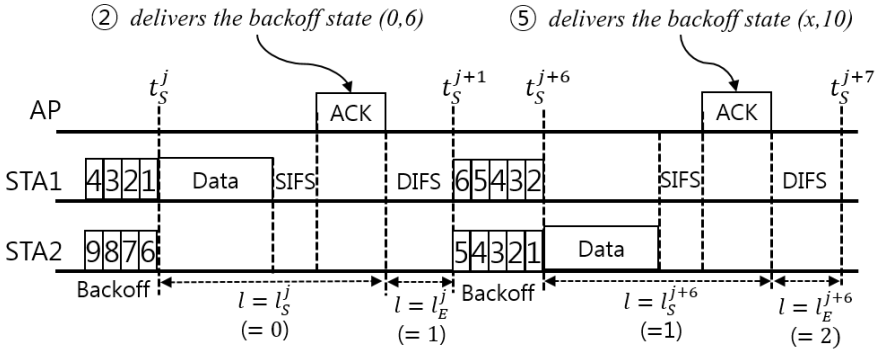
At a later time, the AP then receives a data frame from STA2 (i.e. after step ④ and before step ⑤ in Fig. 3a), the AP uniformly selects a random number from the range [0, 15] *(virtual backoff stage* 0*)* to generate a backoff state for STA2. At this point of time, the AP knows that STA1 has backoff count 1. This internal operation of the AP can be described by the state transitions denoted by ⓐ in Fig. 3c, where $W_0 = 16$. If the selected number is different to the backoff count of STA1, then the backoff state is allocated to STA2 (the state transitions denoted by ⓒ in Fig. 3c). In this case the value of backoff stage to be included in the ACK frame is zero. However, if the selected number is the same as the backoff count of another station *(a virtual collision)*, for example the backoff count of STA1 (=1) as shown in Fig. 3a, the AP again picks another random number in the doubled range [0, 31] *(virtual backoff stage* 1*)*. This can be described by the state transitions denoted by ⓑ in Fig. 3c, where $W_1 = 32$. However, if the number selected in [0, 31] is equal to the backoff count of STA1 again, then the AP selects again a random backoff count in the doubled range [0, 63] *(virtual backoff stage* 2*)*. This can be described by the state transitions denoted by ⓓ in Fig. 3c. The process continues to the point where the range is [0, 1023] *(virtual backoff stage* 6*)*, whereupon random numbers are selected in this range until a unique value (a backoff count to be included in the ACK frame) is obtained[10]. We call this a virtual backoff algorithm (VBA) in the AP. In this way, nodes contending at the same time for accessing the channel can all be allocated a unique backoff count. (Fig. 4 presents pseudo-code for the operation of VBA, and Fig. 5 presents the equivalent pseudo-code for operation of a station.) We allow the AP to allocate a unique backoff state to itself (using the VBA based on the synchronized backoff counts) when it has a data frame to send.

According to the IEEE 802.11a standard, although it supports eight different rates in the range from 6 Mbps up to 54 Mbps, only three rates (i.e. 6 Mbps, 12 Mbps, and 24 Mbps)

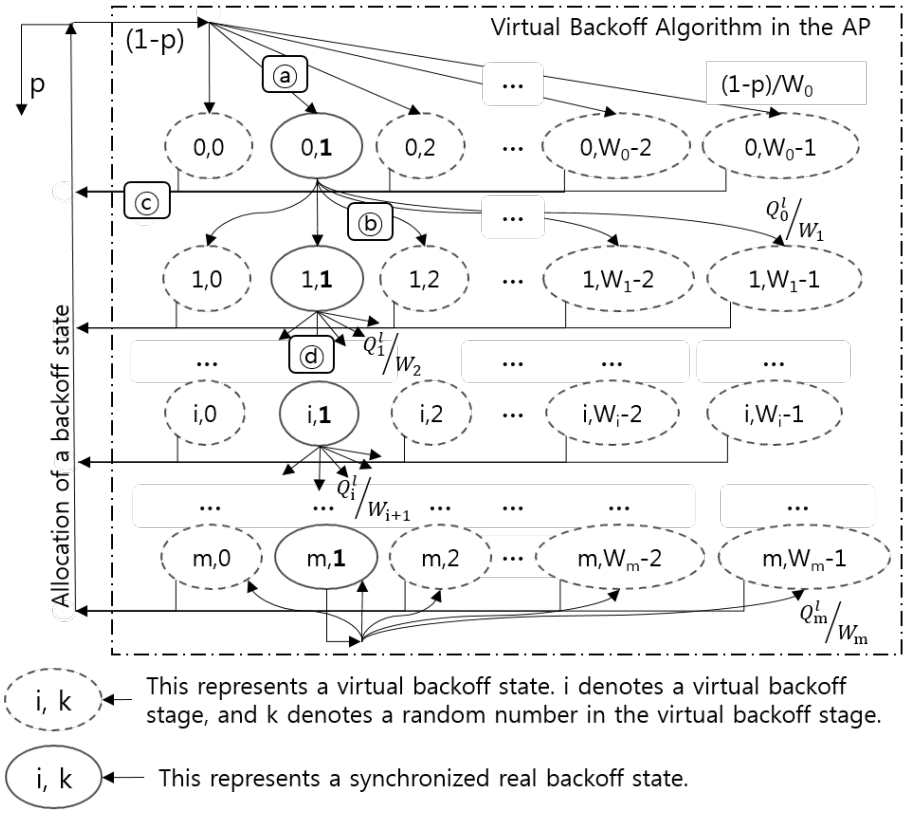
[10]This maximum value, W_m, is chosen to match the operation of DCF in order to maintain fairness, but will limit the BSS to 1024 nodes.



(a) Allocation of a unique backoff count



(b) The flow of frames in CRB. The backoff stage value x in step ⑤ is equal to the number of *virtual collisions* that have occurred during the VBA.



(c) Virtual backoff algorithm (VBA)

Fig. 3. Centralized random backoff (CRB) with virtual backoff algorithm (VBA).

```
1: if a successful reception of a data frame then
2:     i = 0
3:     x = rand(0, 2^i W_0 − 1)
4:     while x is not unique compared to the SBCs (i.e. a virtual collision occurs)
5:         if i < m then
6:             i = i + 1
7:         x = rand(0, 2^i W_0 − 1)
8:     Send the ACK frame with the backoff state (i, x)
9: else
10:    Do not send an ACK frame
```

Fig. 4. The pseudo-code of VBA.

```
1: if Tx queue was empty, before a packet has arrived from upper layers then
2:     if channel is sensed idle then
3:         Start transmitting immediately
4:     else channel is sensed busy
5:         i = 0
6:         x = rand(0, 2^i W_0 − 1)
7:         Start backoff procedure with the state (i, x)
8: else (this node has been active, i.e. saturation condition)
9:     if a successful reception of a backoff state (i, k) from the AP then
10:        Start backoff procedure with the state (i, k)
11:    else
12:        if i < m then
13:            i = i + 1
14:        x = rand(0, 2^i W_0 − 1)
15:        Start backoff procedure with the state (i, x)
```

Fig. 5. The pseudo-code of the operation of a station.

are mandatory. This means in order to support backward compatibility to 802.11a devices, one of the three rates has to be used for transmitting control frames such as ACK frames. When the 6 Mbps rate (i.e. BPSK[11] modulation with rate $1/2$ coding) is used for transmitting ACK frames, an OFDM (Orthogonal Frequency-Division Multiplexing) symbol is encoded to carry three octets. As seen in Fig. 2, a legacy ACK frame contains 16 octets in its PSDU (PLCP[12] Service Data Unit), and 6 bits in its tail. Therefore, 6 OFDM symbols are required to carry the PSDU and the tail bits. This means that adding two additional octets for including the CRB field

[11]Binary Phase-Shift Keying

[12]Physical Layer Convergence Protocol

in a legacy ACK frame will require an additional OFDM symbol. However, if the 24 Mbps rate (i.e. 16-QAM[13] with rate $1/2$ coding) is used for transmitting ACK frames, an OFDM symbol is encoded to carry 12 octets. This means that adding two additional octets for including the CRB field in a legacy ACK frame will not require an additional symbol[14].

IV. NUMERICAL ANALYSIS

We consider a single BSS Wi-Fi network consisting of n active (contending) nodes. We assume ideal channel conditions, and there is no hidden node[15]. This assumption implies that no ACK frames are lost after successfully receiving a data frame.

Under these conditions, we first derive a numerical solution for the probability of a virtual collision in VBA. Second, we present a simplified Markov chain model for analysing the probability of a backoff state of a node operating in CRB. Based on this, we obtain the transmission probability (=τ) and the (real) collision probability (=p). Third, using an absorbing Markov chain model, we explain how the number of SCNs changes over time. Lastly, the throughput performance of CRB is analysed and compared to that of a deterministic backoff mechanism.

A. Probability of a Virtual Collision

A node starts its backoff procedure by setting its backoff count by either uniformly choosing a random value from a contention window (after a transmission failure) or receiving a backoff state value from the AP (after a transmission success). The $(m+1)$ backoff stages and the associated $(m+1)$ contention windows can be represented by Fig. 6. In addition, for simplicity of the analysis, we define $(m+1)$ *Ranges* as shown in the figure. Range 0 means the range of integers [0, $W_0 - 1$], and Range i is the range [W_{i-1}, $W_i - 1$] where $(1 \leq i \leq m)$.

We consider the jth slot time[16], which starts at time t_S^j (where $t_S^j < t_S^{j+1}$), and time t_E^j represents the end of the jth slot time, i.e. $t_E^j = \lim_{\epsilon \to 0}(t_S^{j+1} - \epsilon)$. We assume that the backoff states of all nodes are updated between the time t_E^j and the time t_S^{j+1}.

[13]Quadrature Amplitude Modulation

[14]We use 6 Mbps constant rate to transmit ACK frames in evaluation. Although use of the 6 Mbps (or the 12 Mbps) requires one additional OFDM symbol, the impact of the additional OFDM symbol is very small compared to the significant throughput gain from using CRB.

[15]A node is a hidden node if another node in the same BSS cannot hear the first when it communicates with the AP.

[16]The term *slot time* in this paper is the time period of a time slot used in [2].

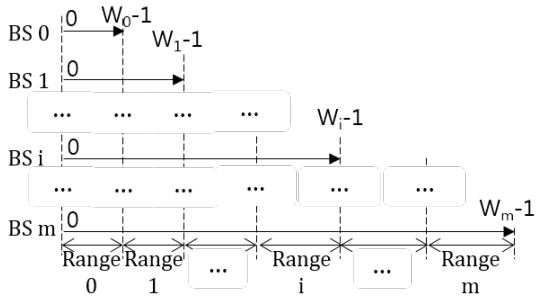


Fig. 6. Backoff stages, contention windows, and ranges.

We define l as the number of SCNs. (Since a SCN has only one SBC, the value l is equal to the number of SBCs.) In addition, we define the terms l_S^j and l_E^j as the number of SCNs at time t_S^j and t_E^j, respectively. (For example, these two notations can be found in Fig. 3b.) This means the number of UCNs at time t_S^j and t_E^j are equal to $(n - l_S^j)$ and $(n - l_E^j)$, respectively.

Since each of the SCNs has a unique SBC with respect to each other, only one of l_E^{j-1} SCNs can possibly start to transmit at time t_S^j. If one of the SCNs starts to transmit at time t_S^j (due to its SBC expiration), then $l_S^j = l_E^{j-1} - 1$. If the transmission is successful, then the source node receives a backoff state from the AP (i.e. $l_E^j = l_S^j + 1$); otherwise, the node (independently) generates a backoff state by itself and it becomes an UCN (i.e. $l_E^j = l_S^j$). In addition, if one of the UCNs starts to transmit at time t_S^j (due to its UBC expiration) and the transmission is successful, then the source node receives a backoff state from the AP (i.e. $l_E^j = l_E^{j-1} + 1$). In this way, the values l_S^j and l_E^j (dynamically) vary over time.

Now, we define five variables in order to analyse the distribution of SBCs in the range $[0, W_m - 1]$ when the AP internally generates a new backoff state to include in the ACK frame to be transmitted. First, the scalar N_i^l is defined as the number of SBCs in Range i when the total number of SCNs is l. For example, suppose that there are three SCNs, and the values of the SBCs are 3, 10, and 25. In this case, assuming $W_0 = 16$ and $m = 6$, we see the relations $l = 3$, $N_0^l = 2$, $N_1^l = 1$, and $N_i^l = 0$ given $(2 \leq i \leq m)$. We also see $\sum_{i=0}^{m} N_i^l = l$.

Second, the scalar Q_i^l is defined by equation (1) and is equal to the probability of a virtual collision at virtual backoff stage (VBS) i when the number of SCNs is l.

$$Q_i^l = \frac{\sum_{k=0}^{i} N_k^l}{W_i} \qquad 0 \leq i \leq m \tag{1}$$

For example, suppose that there are three SCNs, and the values of the SBCs are 3, 10, and 25. In this case, we see $Q_0^l = 2/16$ and $Q_i^l = 3/(16 \cdot 2^i)$ where $(1 \leq i \leq m)$. In addition, the

notation Q_i^l can be found in Fig. 3c, where we see $l = 1$, $N_0^l = 1$, $N_i^l = 0$ given $(1 \leq i \leq m)$, and $Q_i^l = 1/W_i$ given $(0 \leq i \leq m)$.

Third, we define the notation P_i^l by equation (2) which means the probability of selecting a unique SBC in VBS i when the number of SCNs is l.

$$P_i^l = \begin{cases} (1 - Q_0^l) & i = 0 \\ (1 - Q_i^l) \prod_{k=0}^{i-1} Q_k^l & 1 \leq i < m \\ \prod_{k=0}^{m-1} Q_k^l & i = m \end{cases} \tag{2}$$

For example, suppose that there are three SCNs, and the values of the SBCs are 3, 10, and 25. In this case, assuming $W_0 = 16$ and $m = 6$, the value of P_0^l (i.e. the probability of selecting a unique SBC in the range of $[0, 15]$) is $14/16$, and the value of P_1^l (i.e. the probability of selecting a unique SBC in the range of $[0, 31]$) is $(29/32) \times (2/16)$.

We assume that the number n is assumed to be less than the value of $(W_m - 1)$ to guarantee that the AP generates a unique SBC at any slot time with a successful data frame transmission. This implies the relation $\sum_{k=0}^{m} P_k^l = 1$.

Fourth, we define the notation Z^l as the probability of selecting zero as a SBC when the number of SCNs is l. For example, suppose there are two SCNs, and the values of the SBCs are 3 and 10. In this case, we see the relations $l = 2$, $Q_i^l = 2/W_i$ given $(0 \leq i \leq m)$, and the value of Z^l is obtained by (3).

$$\begin{aligned} Z^l|_{SBCs=\{3,10\}} &= \sum_{i=0}^{m} (\text{Picking zero at the } \mathrm{i_{th}} \text{ VBS}) \\ &= \frac{1}{16} + \frac{2}{16}\frac{1}{32} + \frac{2}{16}\frac{2}{32}\frac{1}{64} + \frac{2}{16}\frac{2}{32}\frac{2}{64}\frac{1}{128} + \frac{2}{16}\frac{2}{32}\frac{2}{64}\frac{2}{128}\frac{1}{256} + \frac{2}{16}\frac{2}{32}\frac{2}{64}\frac{2}{128}\frac{2}{256}\frac{1}{512} \\ &+ \frac{2}{16}\frac{2}{32}\frac{2}{64}\frac{2}{128}\frac{2}{256}\frac{2}{512}\frac{1}{1024}\left[1 + \frac{2}{1024} + \left(\frac{2}{1024}\right)^2 + \left(\frac{2}{1024}\right)^3 + \cdots\right] \end{aligned} \tag{3}$$

Because zero can be allocated as a backoff count, it is possible for the source node to transmit multiple data frames consecutively without backoff. For example, as described in Fig. 7, backoff count zero can be allocated to the source node in the jth slot time, and then a new non-zero SBC can be picked and allocated to the source node in step ⑥ in the $(j + 1)$th slot time. In this case, the source node transmits two data frames without backoff.

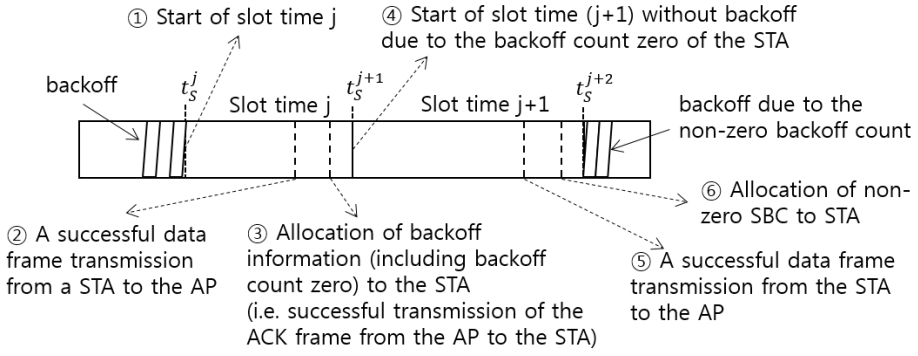


Fig. 7. An example of two consecutive slot times granted to a node

Lastly, we define the scalar D_i^l as the probability of selecting a new non-zero SBC in Range i (i.e. the probability of selecting the $(l+1)$th SBC in Range i) when the number of SCNs is l. The value of D_i^l can be expressed by equation (4),

$$\begin{aligned} D_i^l &= D_i^l|_{(1st\ slot)} + D_i^l|_{(2nd\ slot)} + \cdots = D_i^l|_{(1st\ slot)} + Z^l D_i^l|_{(1st\ slot)} + (Z^l)^2 D_i^l|_{(1st\ slot)} + \cdots \\ &= D_i^l|_{(1st\ slot)} \sum_{j=0}^{\infty} (Z^l)^j = \frac{D_i^l|_{(1st\ slot)}}{1-Z^l} \qquad 0 \le i \le m \end{aligned} \tag{4}$$

where $D_i^l|_{(kth\ slot)}$ represents the probability of selecting a non-zero SBC in Range i in the kth slot time in the series of the consecutive successful slot times. We now see the relation $Z^l = (1 - \sum_{i=0}^{m} D_i^l|_{(1st\ slot)})$, where the term $(\sum_{i=0}^{m} D_i^l|_{(1st\ slot)})$ represents the probability of selecting a non-zero SBC in the first successful slot time.

Using the equation (4) and the result of Appendix A, we obtain the equations in (5). Note that since the value n is assumed to be less than the value of $(W_m - 1)$ to guarantee that the AP can generate a unique non-zero SBC for any series of successful consecutive slot times, we see $\sum_{i=0}^{m} D_i^l = 1$. Although the value of D_i^l is expressed by the two variables N_i^l and Q_i^l in (5), considering equation (1) we see that the value of D_i^l can be expressed by only N_i^l.

$$D_i^l = \begin{cases} \dfrac{(W_0 - N_0^l - 1)}{1-Z^l} \left(\dfrac{1}{W_0} + \displaystyle\sum_{j=0}^{m-2} \left(\dfrac{\prod_{k=0}^{j} Q_k^l}{W_{j+1}} \right) + \dfrac{\prod_{k=0}^{m-1} Q_k^l}{W_m(1-Q_m^l)} \right) & i = 0 \\ \dfrac{(W_{i-1} - N_i^l)}{1-Z^l} \left(\displaystyle\sum_{j=i-1}^{m-2} \left(\dfrac{\prod_{k=0}^{j} Q_k^l}{W_{j+1}} \right) + \dfrac{\prod_{k=0}^{m-1} Q_k^l}{W_m(1-Q_m^l)} \right) & 1 \le i < m \\ \dfrac{(W_{m-1} - N_m^l)}{1-Z^l} \dfrac{\prod_{k=0}^{m-1} Q_k^l}{W_m(1-Q_m^l)} & i = m \end{cases} \tag{5}$$

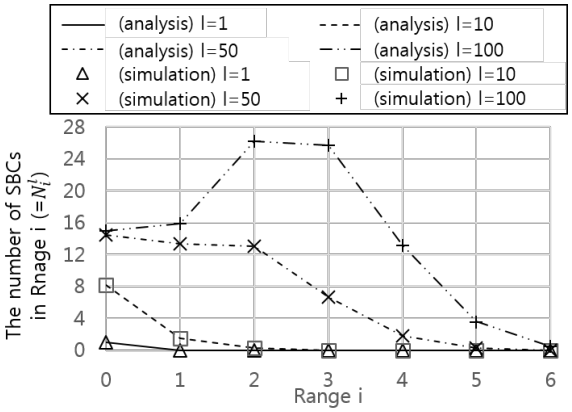


Fig. 8. Analysis and simulation results on N_i^l (i.e. the number of SBCs in Range i when the number of SCNs is l) given $W_0 = 16$ and $m = 6$.

From the definition of the variable D_i^l, we obtain relation (6).

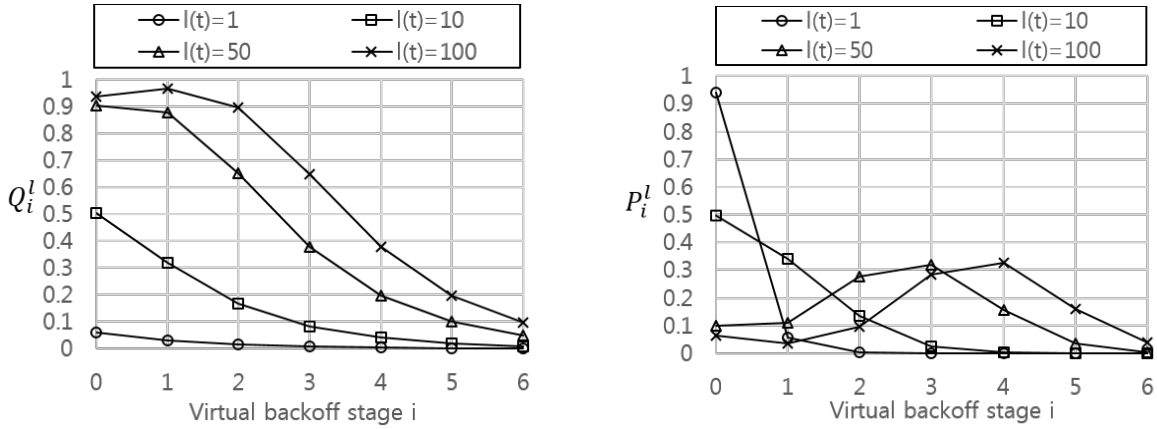
$$N_i^{l+1} = \sum_{k=0}^{l} D_i^k \tag{6}$$

Since the value of D_i^l is expressed by N_i^l (according to the equations (1) and (5)), equation (6) means that the value of N_i^{l+1} can be calculated iteratively. In order to solve equation (6), we assume an initial condition that there initially *was* a SBC in Range 0 (i.e. $N_i^{l=0} = 1$ when $i = 0$ and $N_i^{l=0} = 0$ when $i \in [1, m]$).

In order to see the distribution of SBCs in the range $[0, W_m - 1]$ for a given l, we computed equation (6) with different values of l. In Fig. 8, we see that at a given Range i, the value of N_i^l increases as l increases. In addition, when $l \leq 50$, the value of N_i^l tends to decrease as the Range i increases. This result shows that how SBCs are distributed over the seven different ranges. Moreover, in Fig. 8, we see that the values of N_i^l obtained by simulation[17] are almost identical to that of the analysis results.

The values Q_i^l and P_i^l also change, as the value l varies. In Fig. 9a, we see that at a given VBS i, the value of Q_i^l increases as l increases. Using the calculated values of N_i^l with the initial condition, we also obtain the value of P_i^l as shown in Fig. 9b. The calculated value of P_i^l at a given l is used for Markov chain model analysis for calculating the transmission probability (=τ) of a node operating in CRB.

[17]We have developed a simple program (written in C language) to simulate VBA. The values of the simulation result are average values obtained through one million repetitions.



(a) The probability of a virtual collision at VBS i (b) The probability of selecting a unique SBC in VBS i

Fig. 9. Analysis results on Q_i^l and P_i^l ($W_0 = 16$ and $m = 6$).

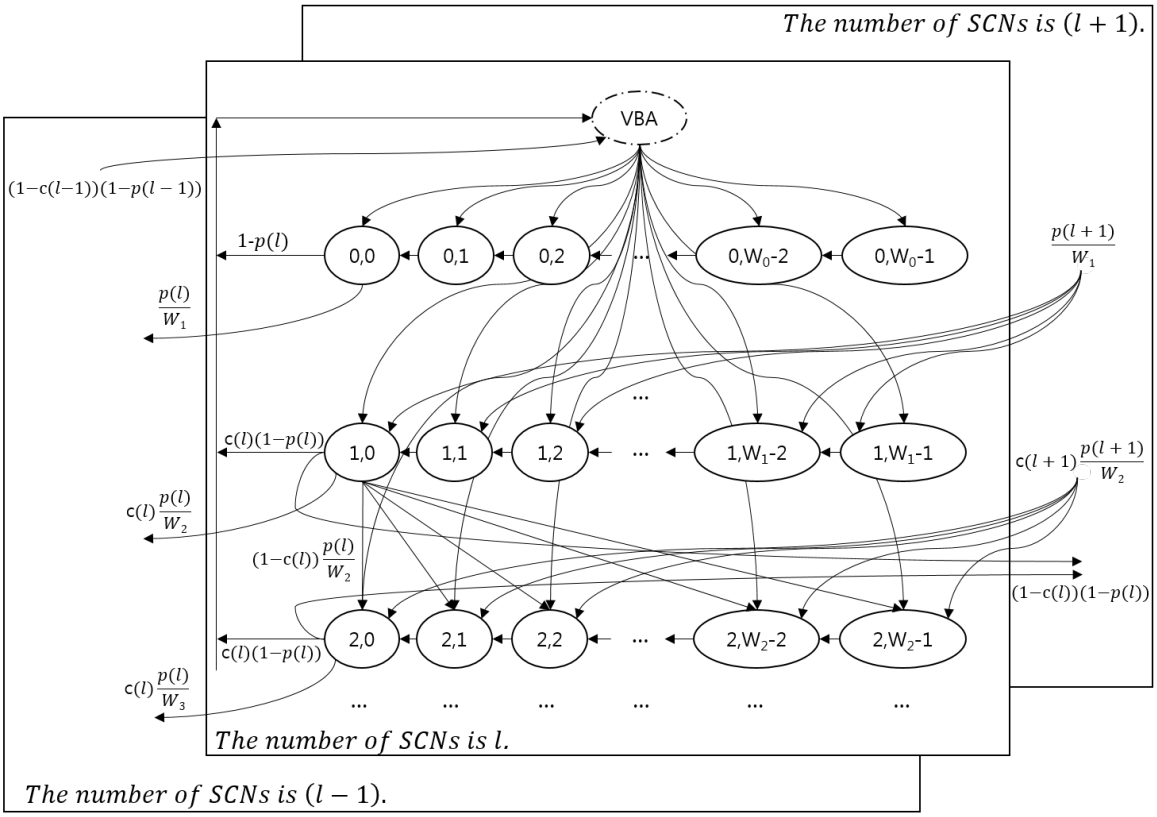


Fig. 10. Markov chain representing the network states during a convergence time period.

B. Probability of a Collision

The network states during a convergence time period can be represented by Fig. 10, where the notation $p(l)$ denotes the collision probability, and the notation $c(l)$ represents the probability of a node being synchronized. Since it is too complex to enable a closed form solution to be

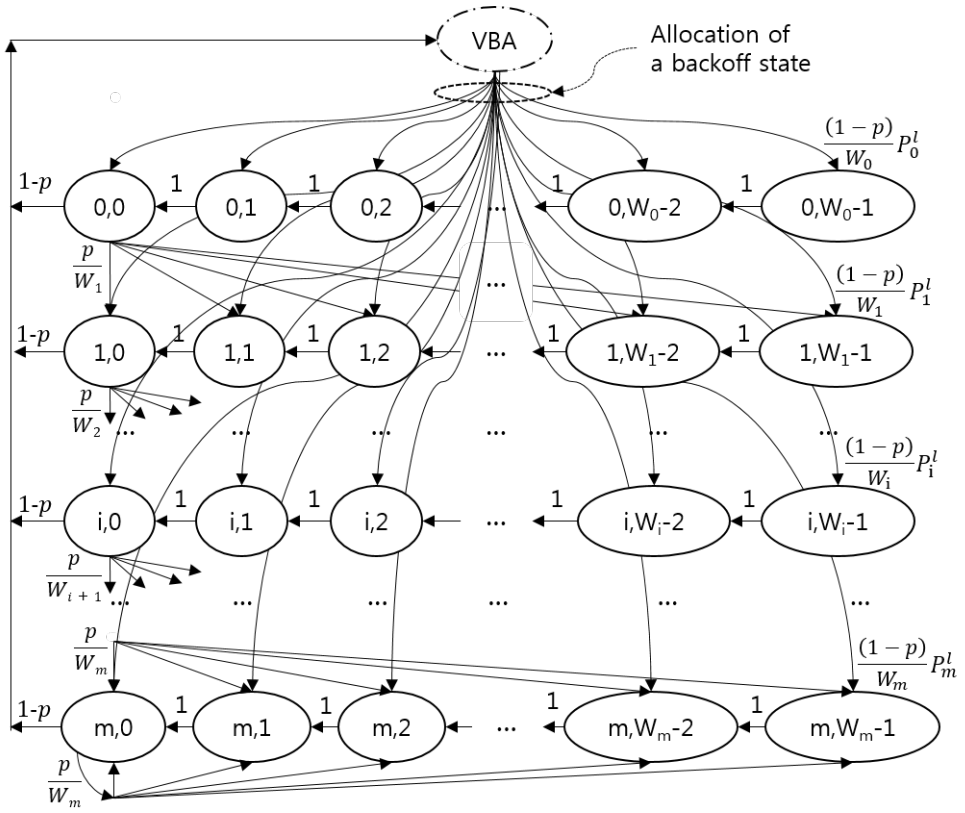


Fig. 11. Markov chain models of a node operating in CRB. When $l \in [0, n-1]$, the model represents a collision prone state where $(0 < p < 1)$. When $l = n$, the model represents a collision free state where $p = 0$.

found, we propose a simplified chain model shown in Fig. 11.

We assume that the value P_i^l is given at a value l (i.e. not depending on time), and find the values τ and p from Fig. 11. The chain model illustrates an internal backoff state of a node. After a successful transmission, the node starts its backoff procedure with the allocated backoff state. The process of allocating the backoff state is illustrated by the curved lines in Fig. 11. When the value l varies in the range [1, $n-2$], the notation p (i.e. $p(l)$ in Fig. 10) implicitly includes an approximation. The approximation can be expressed by equation (7).

$$\begin{cases} p(l-1) \simeq p(l) \\ p(l+1) \simeq p(l) \end{cases} \tag{7}$$

In addition, when $l = n-1$, we assume that $p(l-1) \simeq p(l)$ and $p(l+1) = 0$.

We define the probability of a backoff state $b_{i,k}$ (where $i \in [0, m]$ and $k \in [0, W_i - 1]$) as

equation (8),

$$b_{i,k} = \lim_{N\to\infty} \frac{1}{N} \sum_{n=1}^{N} P\{s_n(t) = i, b_n(t) = k\} \tag{8}$$

where $s_n(t)$ and $b_n(t)$ denote the stochastic process representing the backoff stage and the backoff counter respectively for the n-th independent realization at time t.

The transition probabilities in the Markov Chain model shown in Fig. 11 are given by the equations from (9a) to (9d),

$$P\{i,k|i,k+1\} = 1 \qquad k \in [0, W_i - 2], i \in [0, m] \tag{9a}$$

$$P\{j,k|i,0\} = \frac{(1-p)}{W_j} P_j^l \qquad k \in [0, W_j - 1], i \in [0, m], j \in [0, m] \tag{9b}$$

$$P\{i,k|i-1,0\} = \frac{p}{W_i} \qquad k \in [0, W_i - 1], i \in [1, m] \tag{9c}$$

$$P\{m,k|m,0\} = \frac{p}{W_m} \qquad k \in [0, W_m - 1] \tag{9d}$$

where $P\{i_1,k_1|i_0,k_0\}$ denotes the probability of a backoff state transition from $\{i_0,k_0\}$ to $\{i_1,k_1\}$. Equation (9a) shows that the BC is decreased at the beginning of each slot time. The second equation (9b) represents the fact that a new packet following a *successful packet transmission* starts backoff with the allocated backoff state. Equations (9c) and (9d) model the state transition after an *unsuccessful transmission*. Equation (9c) shows that when an unsuccessful transmission occurs at BS $(i-1)$, the BS increases by one, and a new BC is uniformly and independently chosen in the range [0, $W_i - 1$]. Equation (9d) models the fact that once the BS reaches the value m, a node stays in the BS m until a successful packet transmission.

From the Markov chain in Fig. 11, the transmission probability can be represented by (10).

$$\tau = \sum_{i=0}^{m} b_{i,0} = \frac{b_{0,0}}{(1-p)P_0^l} \tag{10}$$

If the value of l is zero, then $P_{i=0}^l = 1$ and $P_i^l = 0$ where $i \in [1, m]$. In this case, the Markov Chain model in Fig. 11 becomes identical to the Markov chain model presented in [2], and the equations (9) and (10) also become identical to those of legacy nodes presented in [2].

In Appendix B, we obtain $b_{0,0}$ as (11),

$$b_{0,0} = \frac{1}{\frac{W_0+1}{2} + \sum_{i=1}^{m-1}\left[\frac{W_i+1}{2}(p^i + \sum_{j=0}^{i-1} f_i^j(l))\right] + \frac{W_m+1}{2}\left(\frac{p^m}{1-p} + \sum_{j=0}^{m-1}\frac{f_m^j(l)}{(1-p)}\right)} \tag{11}$$

where $f_i^j(l) = (p^{i-1-j})P_{j+1}^l/P_0^l$ for short. We see in equation (11) that $b_{0,0}$ is a function of p and P_i^l. This means that by substituting the value of $b_{0,0}$ in the equation (10) with the equation (11), the value of τ is given as a function of p and P_i^l. This is one relation between τ and p given both l and n. In order to find τ and p at given l and n, we need another equation relating τ and p.

Let $P_{tr}(l)$ be the probability that there is at least one node starting to transmit in a considered slot time when the number of SCNs is l. We obtain $P_{tr}(l)$ as (12),

$$P_{tr}(l) = 1 - (1 - P_{tr}^{un}(l))(1 - P_{tr}^{sn}(l)) \tag{12}$$

where the notation $P_{tr}^{un}(l)$ represents the probability that there is at least one UCN starting to transmit. The notation $P_{tr}^{sn}(l)$ represents the probability that there is at least one SCN starting to transmit. This probability depends on the number of SBCs in Range 0 (i.e. N_0^l). For example, as the value N_0^l becomes close to the value $(W_0 - 1)$, the SCNs will transmit in consecutive time slots. Since the values of SBCs decrease as time increases, the SBCs in Range 1 (i.e. N_1^l) will move to Range 0 before the backoff counter reaches zero. The values of $P_{tr}^{un}(l)$ and $P_{tr}^{sn}(l)$ can be obtained by (13),

$$\begin{cases} P_{tr}^{un}(l) = 1 - (1-\tau)^{n-l} \\ P_{tr}^{sn}(l) = 1 - \left(1 - \dfrac{N_0^l}{W_0 - 1}\right)(1 - P_{tr}P_s Z^l) \end{cases} \tag{13}$$

where the term $[N_0^l/(W_0 - 1)]$ in the second equation in (13) represents the density of SBCs in Range 0, and the term $[1 - N_0^l/(W_0 - 1)]$ means the density of non-allocated numbers in Range 0. The term $(1 - P_{tr}P_s Z^l)$ denotes the probability of allocating a non-zero BC.

We define $P_s(l)$ as the probability that a transmission occurring on the channel is successful when the number of SCNs is l. This is equal to the probability that *exactly one station* transmits on the channel, conditioned on the fact that *at least one station* transmits. This yields equation

(14),

$$P_s(l) = P_s^{un}(l) + P_s^{sn}(l) \tag{14}$$

where the notations $P_s^{un}(l)$ and $P_s^{sn}(l)$ represent the probability of a successful slot time with a packet sent by an UCN and a SCN, respectively. The values of $P_s^{un}(l)$ and $P_s^{sn}(l)$ are obtained by equations in (15).

$$\begin{cases} P_s^{un}(l) = \dfrac{(n-l)\tau(1-\tau)^{n-l-1}(1-P_{tr}^{sn}(l))}{P_{tr}(l)} \\ P_s^{sn}(l) = \dfrac{P_{tr}^{sn}(l)(1-P_{tr}^{un}(l))}{P_{tr}(l)} \end{cases} \tag{15}$$

Note that if the value of l is zero, then $P_{tr}^{sn}(l) = 0$, $P_s^{sn}(l) = 0$, and the expression of $P_s(l)$ becomes identical to that of legacy nodes presented in [2].

The probability p that a packet encounters a collision is equal to the probability that at least one of the $(n-1)$ remaining stations starts to transmit in the considered slot time. If one UCN starts transmitting a packet in a considered slot time, then there will be no collision if neither the $(n-l-1)$ UCNs nor the l SCNs start transmitting at the same time. In the case that a SCN starts transmitting a packet, since each of the SCNs has a unique backoff count to each other, there will be no collision provided that the $(n-l)$ UCNs do not start transmitting at the same time. This yields the two equations in (16),

$$\begin{cases} p^{un}(l) = 1 - (1-\tau)^{n-l-1}(1-P_{tr}^{sn}(l)) \\ p^{sn}(l) = 1 - (1-\tau)^{n-l} \end{cases} \tag{16}$$

where the notation $p^{un}(l)$ and $p^{sn}(l)$ represents the probability of a collision seen by a packet transmitted by an UCN and a SCN, respectively.

Taking the average between $p^{un}(l)$ and $p^{sn}(l)$, we obtain the value of p as (17),

$$p = \frac{(n-l)\tau}{(n-l)\tau + P_{tr}^{sn}(l)} p^{un}(l) + \frac{P_{tr}^{sn}(l)}{(n-l)\tau + P_{tr}^{sn}(l)} p^{sn}(l) \tag{17}$$

where the denominator $[(n-l)\tau + P_{tr}^{sn}(l)]$ represents the total number of packets transmitted in the considered slot time. This equation is the second relationship between τ and p, and now we can obtain τ and p given both l and n.

Table I. The value of Δl between two consecutive slot times

Tx node(s)	Tx result	Probability		Δl
		Symbol	Value	
None (empty slot)	-	$P_0(l)$	$1 - P_{tr}(l)$	0
Only one UCN	S	$P_1(l)$	$P_{tr}(l)P_s^{un}(l)$	$+1$
Only multiple UCNs (i.e. No SCN)	F	$P_2(l)$	$P_{tr}^{un}(l)(1 - P_{tr}^{sn}(l))$ $-P_{tr}(l)P_s^{un}(l)$	0
Only one SCN	S	$P_3(l)$	$P_{tr}(l)P_s^{sn}(l)$	0
UCN(s) and one SCN	F	$P_4(l)$	$P_{tr}^{un}(l)P_{tr}^{sn}(l)$	-1

* S for success, F for failure, and l for l_E^{j-1}.

C. Number of Synchronized Nodes

All nodes are assumed to be using the CRB protocol, and the number of SCNs (=l) dynamically varies in the range $[0, n]$. We define a vector P^j as the probability distribution of the value l at slot time j, which can be expressed by (18).

$$P^j = \begin{bmatrix} p_0^j & p_1^j & \cdots & p_{n-1}^j & p_n^j \end{bmatrix} \tag{18}$$

Each element p_i^j (where $i \in [0, n]$) represents the probability that the value of l is equal to i at slot time j. This means $\sum_{i=0}^{n} p_i^j = 1$. We find the distribution vector P^j and investigate how the distribution vector P^j changes over slot times.

Now, we define Δl as the gap between l_E^j and l_E^{j-1}, i.e. $\Delta l = l_E^j - l_E^{j-1}$. The possible values of Δl are now described. First, $\Delta l = 1$ if the jth slot time was successful with a transmission from one of UCNs (i.e. the UCN has been synchronized after the slot time). Second, $\Delta l = -1$ if a collision with a packet sent by a SCN was occurred in the slot time (i.e. the SCN has been unsynchronized). Lastly, $\Delta l = 0$ in the three cases: an empty slot time, a collision among UCNs, and a successful transmission by a SCN. These five different cases are summarized in Table I. The sum of the five different probabilities in the table is one for every slot time, i.e. $\sum_{k=0}^{4} P_k(l) = 1$ where $(1 \leq j)$. Table I shows the value of Δl for each of the five different cases.

The absorbing Markov chain model depicted in Fig. 12 illustrates a state of the wireless network in terms of the number of SCNs. Note that the time scale of the chain model in Fig. 11 is a considered time slot, while the time scale of the absorbing chain model in Fig. 12 is the period during which the network moves from the initial state to a collision free state. The notation $S_{x,y}$ in Fig. 12 presents the probability of a state transition from the state that $l = y$ to the state

that $l = x$. The state that $l = 0$ (i.e. where all the nodes are unsynchronized) is identical to the state of wireless network operating in the legacy DCF protocol. As denoted in Fig. 12, if the value of l is equal to n, then the state becomes a collision free state (which is an absorbing state). This means $S_{n,n}$ is always one. The absorbing Markov chain can be expressed by (19).

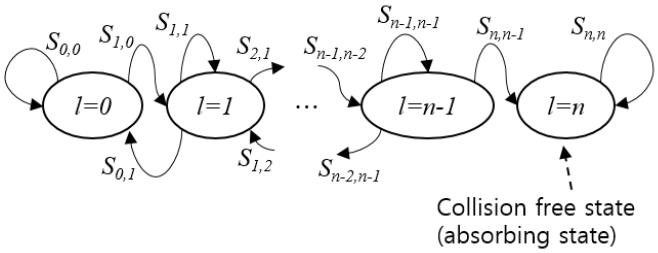


Fig. 12. Markov chain model for analysis of the vector P^j.

$$A = \begin{bmatrix} S_{0,0} & S_{1,0} & 0 & \cdots & 0 & 0 \\ S_{0,1} & S_{1,1} & S_{2,1} & \cdots & 0 & 0 \\ 0 & S_{1,2} & S_{2,2} & \cdots & 0 & 0 \\ 0 & 0 & S_{2,3} & \cdots & 0 & 0 \\ \cdots & & & \cdots & \cdots & \\ 0 & 0 & 0 & \cdots & S_{n-1,n-1} & S_{n,n-1} \\ 0 & 0 & 0 & \cdots & 0 & S_{n,n} \end{bmatrix} \tag{19}$$

From Table I, we obtain relations in (20).

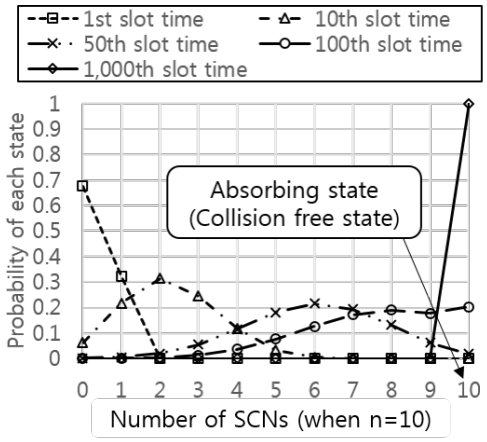
$$S_{l,l} = P_0(l) + P_2(l) + P_3(l) \qquad l \in [0, n-1] \tag{20a}$$

$$S_{n,n} = 1 \qquad l = n \tag{20b}$$

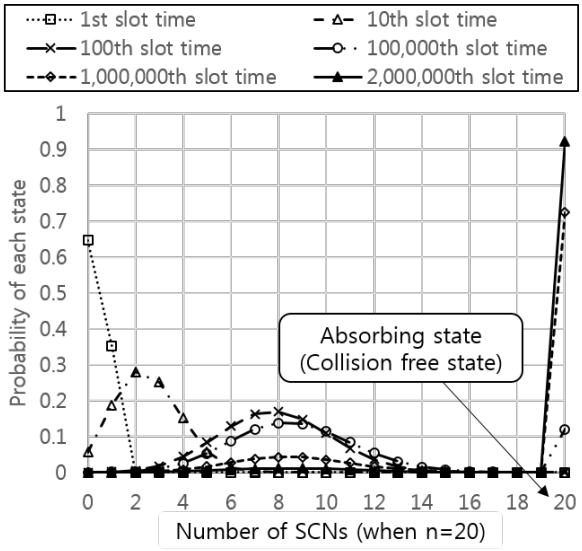
$$S_{l+1,l} = P_1(l) \qquad l \in [0, n-1] \tag{20c}$$

$$S_{l-1,l} = P_4(l) \qquad l \in [1, n-1] \tag{20d}$$

Now, the vector P^j can be obtained by equation (21), where the vector I represents the initial state of the network. We assume that the value l is zero when $j = 0$ (i.e. all the nodes are assumed to be randomized/unsynchronized nodes at the beginning of the first slot time). This



(a) Vector P^j when n=10



(b) Vector P^j when n=20

Fig. 13. Analysis results of vector P^j ($W_0 = 16$ and $m = 6$).

means the probability of the network state where $l = 0$ is one when $j = 0$.

$$P^j = IA^j \quad \text{where} \quad I = \begin{bmatrix} 1 & 0 & \cdots & 0 & 0 \end{bmatrix} \tag{21}$$

Fig. 13 shows the calculated results for the distribution vector P^j using equation (21) when $n = 10$ and $n = 20$. First of all, we see that the probability of the state $l = 10$ in Fig. 13a increases to 1, as the slot time increases up to 1000. Moreover, we see that the probability of the state $l = 20$ in Fig. 13b increases to 1, as the slot time increases over 2,000,000. In this result, we see an important result that the wireless network operating in CRB moves toward a collision free state without adjusting/tuning the contention window size for the given number n. However, we also see that the number of slot times required to move toward a collision free state dramatically increases as the number of nodes n increase.

D. Throughput Performance

We reuse the definition of saturation throughput (=S) in [2], which can be represented by equation (22).

$$S = E[\text{payload}]/E[\text{slot time}] \tag{22}$$

We compute the throughput at slot time j (=S^j) as in (23), where $(P^j)^T$ represents the transpose of vector P^j. Each element S_i of the vector C denotes the saturation throughput when l is

Table II. Parameters used to obtain numerical analysis results

Parameters	Value	Parameters	Value	Parameters	Value
Bit rate for Data frames	54 Mbps	Empty slot time	9 μs	Preamble signal duration	16 μs
Bit rate for ACK frames	6 Mbps	SIFS time	16 μs	PLCP header duration	4 μs
UDP payload	1400 bytes	DIFS time	34 μs	W_0	16
UDP header + IP header	28 bytes	MAC header	34 bytes	W_m	1024

assumed to be i.

$$S^j = C \cdot (P^j)^T \quad \text{where} \quad C = \begin{bmatrix} S_0 & S_1 & \cdots & S_{n-1} & S_n \end{bmatrix} \tag{23}$$

A successful transmission occurs in a slot time with probability $P_{tr}(l)P_s(l)$. The slot time is empty with probability $(1 - P_{tr}(l))$. The slot time contains a collision with probability $P_{tr}(l)(1 - P_s(l))$. Therefore, the value S_l can be obtained by (24),

$$S_l = \frac{P_{tr}(l)P_s(l)E[P]}{(1 - P_{tr}(l))\sigma + P_{tr}(l)P_s(l)T_s + P_{tr}(l)(1 - P_s(l))T_c} \tag{24}$$

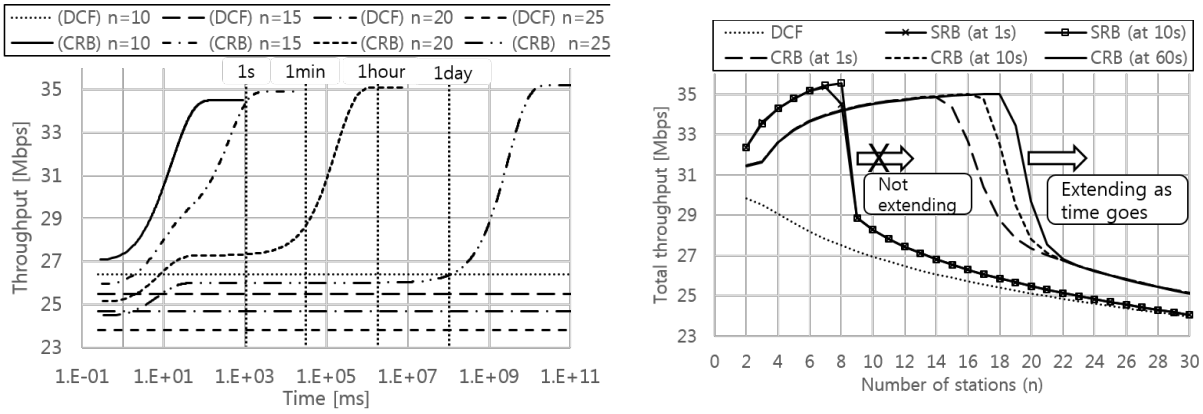
where T_s denotes the average time the channel is sensed busy because of a successful transmission, and T_c represents the average time the channel is sensed busy due to a collision. The notation σ represents the duration of an empty slot time. The scalar $E[P]$ denotes the average packet payload size successfully transmitted.

The values T_s and T_c are given by (25),

$$\begin{aligned} T_s &= H + E[P] + SIFS + \delta + ACK + DIFS + \delta \\ T_c &= H + E[P^*] + DIFS + \delta \end{aligned} \tag{25}$$

where $H(= PHY_{hdr} + MAC_{hdr})$ denotes the packet header, δ represents the propagation delay, and $E[P^*]$ is the average length of the longest packet payload involved in a collision. In the case all packets have the same fixed size, the value $E[P^*]$ becomes equal to the value $E[P]$. The notation *SIFS* (Short Inter Frame Space) denotes the delay time required for a wireless interface to process a received frame and to respond with a response frame. The scalar *ACK* is the transmission time of an ACK frame. The notation *DIFS* (DCF Inter Frame Space) represents the standard wait time required before starting the random backoff procedure in the saturation condition.

The parameters given in Table II were used to obtain analysis results shown in Fig. 14. We see in Fig. 14a that the throughput converges to a maximum value as the time increases. The



(a) Throughput versus time

(b) Throughput versus the number of nodes

Fig. 14. Analysis results on total throughput.

figure also shows that similar to Fig. 13 the time period required for convergence dramatically increases, as the number of n increases. The maximum throughput value achieved in a collision free state is slightly increased as the value n increases. This is because the probability of an empty slot time decreases, while the probability of a collision is zero.

We see in Fig. 14b that the throughput performance when using CRB changes over time (i.e. non-stationary state). The figure shows that given a sufficient time, the wireless network with a larger number of nodes does reach a collision free state. For example, the figure shows that 14 nodes can reach the collision free state within one second without tuning the contention window size (i.e. $W_0 = 16$ and $m = 6$). This is a significant advantage of CRB over SRB, because according to [20] the maximum number of SRB nodes that can converge to a collision free state (given $W_0 = 16$) is limited to 8. However, the figure also shows that it takes an hour for 20 nodes to converge to the collision free state. In practice we anticipate that the performance of a heavily loaded network will be between that of SRB and the optimum CRB[18].

V. SIMULATION AND NUMERICAL RESULTS

In order to validate the numerical analysis, we have developed a network simulator based on the Network Simulator-3 (NS-3)[19]. The *NS-3 Wi-Fi module* [25] has been modified to implement the CRB protocol. The parameters used in simulation are summarized in Table III.

Fig. 15 shows four different simulation scenarios. First, in a simulation setup shown in Fig. 15a, there is no hidden node, and the channel condition is assumed to be perfect (i.e.

[18]To reduce the convergence time when the value n is large a re-synchronization process can be used. The AP simply transmits a new unique backoff count to each station, which all the stations start to use at a specified time.

[19]NS-3 version 3.25. This is an open source project available at http://www.nsnam.org.

Table III. Simulation parameters

Parameters	Value	Parameters	Value	Parameters	Value
Wireless standard	IEEE 802.11a PHY	Bandwidth	20 MHz	Frequency channel	5.0 GHz
Transmission queue	Single queue	Traffic model	Full buffer	UDP payload	1400 bytes
Propagation loss model	Log distance	W_0	16	Rx sensitivity	-91 dBm
Available data rates	6, 9, 12, 18, 24, 36, 48, and 54 Mbps	W_m	1024	Transport layer	UDP
		Tx power	16 dBm	BER model	Reference [25]

* BER: Bit Error Rate

no frame errors caused by channel fading or other effects). Random values selected from a triangular distribution are used for the distances from the AP to each of the stations, which is consistent with a circular coverage region. Second, Fig. 15b shows a simulation setup for testing backward compatibility to the DCF protocol[20]. In this paper, backward compatibility means an improvement of total throughput performance of the wireless network where the nodes operating in CRB coexist fairly with the nodes operating using DCF. Third, using the simulation setup presented in Fig. 15c, we observe the effects of hidden nodes on the performance of CRB. Each of the stations in group 1 in Fig. 15c can neither decode nor sense carrier signals from the stations in group 2, and vice versa. Lastly, Fig. 15(d) shows the simulation configuration for testing CRB in two overlapped APs.

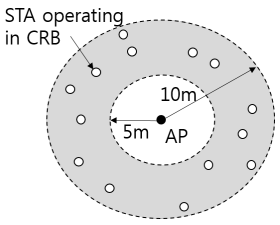
A. CRB nodes without hidden nodes

Fig. 16 shows simulation results on total throughput obtained for the setup shown in Fig. 15a. In this case, a 54 Mbps constant rate was used for transmitting data frames, and a 6 Mbps constant rate was used for transmitting ACK frames. Because of the network needing to settle into a collision free state, only statistics obtained for the last 0.2 second of each repeated simulation were used to draw simulation results[21]. Fig. 16a shows that the total throughput when using CRB outperforms that of using DCF, as the offered load[22] per station increases above 2.7 Mbps. Fig. 16b shows that when $n = 16$, the total throughput of CRB increases from 32.4 Mbps to 34.8 Mbps as the simulation time increases from 1 second to 60 seconds. In addition, we see

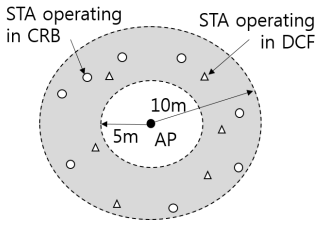
[20]We assume that there is an additional exchange of information between the AP and each station in connection establishment procedure (e.g. exchange of *CRB support bit*). By doing so, the AP can use the legacy ACK frame format (i.e. not including CRB field) for legacy nodes, while it uses the proposed ACK frame format (including CRB field) for nodes supporting CRB.

[21]We calculate average values (e.g. average total throughput and average retransmission ratio per packet) by monitoring data frames transmitted for the last 0.2 s. The number of the data frames is larger than 500 in using the parameters in Table III.

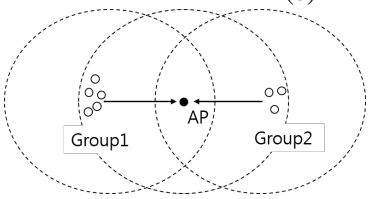
[22]In this paper, *offered load* means UDP payloads generated by the on/off application per second, and *throughput* means successfully received MAC service data units (MSDUs) per second. The MSDUs include a UDP header and an IP header.



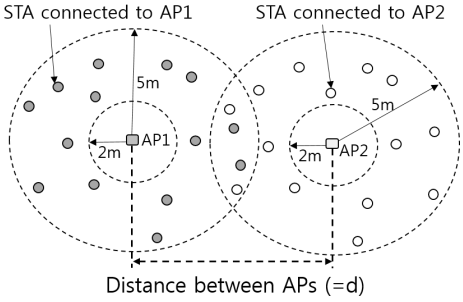
(a) CRB nodes without hidden nodes



(b) Mixed nodes without hidden nodes

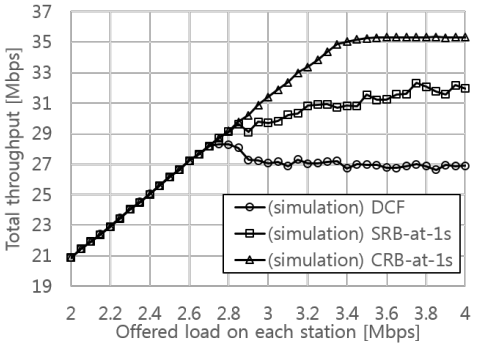


(c) CRB nodes with hidden nodes

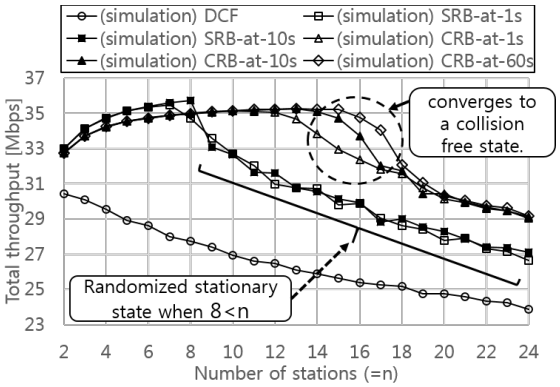


(d) Two overlapped APs

Fig. 15. Two-dimensional simulation setups.



(a) Offered load per station versus throughput (n=10)



(b) Number of stations versus total throughput

Fig. 16. Simulation results on the throughput of CRB, SRB, and DCF in the single AP setup without hidden nodes ($W_0 = 16$ and $W_m = 1024$).

that when the value n is larger than 8, the throughput performance when using CRB outperforms that of using SRB.

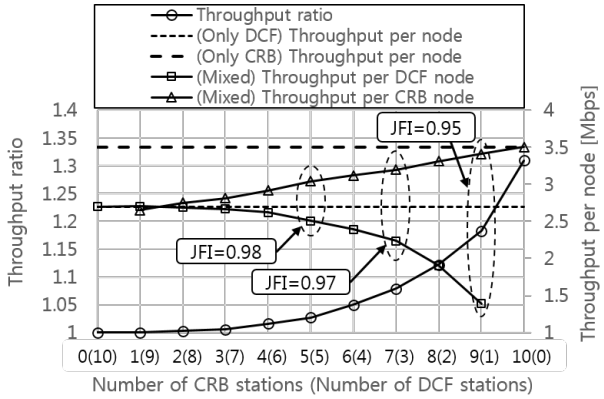


Fig. 17. Simulation results in the mixed nodes scenario ($n = 10$).

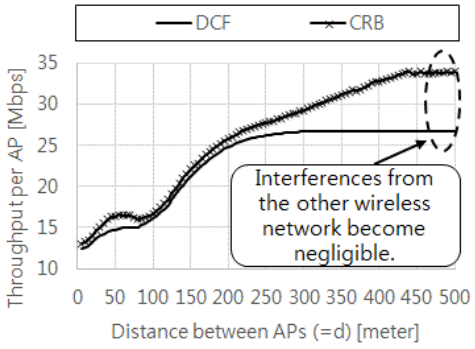


Fig. 18. Simulation results in the two overlapped APs setup (n per each AP= 10).

B. Mixed nodes without hidden nodes

Fig. 17 shows simulation results for the mixed node setup shown in Fig. 15b. In this simulation, a 54 Mbps constant rate was used for transmitting data frames, and a 6 Mbps constant rate was used for transmitting ACK frames. In Fig. 17, throughput ratio (which means the total throughput of the mixed network compared to that of the legacy DCF network) shows that as the proportion of CRB nodes increases, the total throughput increases. For example, when five nodes operating in CRB coexist with five nodes using DCF (i.e. the case 5(5) on the x-axis), the total throughput gain is about 3% compared to that of when all ten nodes use DCF (i.e. the case 0(10)). However, when eight nodes operating in CRB coexist with two nodes using DCF (i.e. 8(2)), the total throughput gain is about 12%. In addition, Fig. 17 shows the throughput per each node operating in the mixed network. While the throughput per CRB node increases with the proportion of CRB nodes, the throughput per DCF node decreases. This is because as the proportion of CRB nodes increases, the DCF nodes tend to have a higher collision probability than the CRB nodes. However, regardless of the proportion of CRB nodes, the Jain's fairness index (in terms of average throughput per node) of the mixed network is maintained above 0.95.

C. In the presence of hidden nodes

We performed simulations in the hidden node setup shown in Fig. 15c. In this simulation, an 18 Mbps constant rate was used for transmitting data frames, and a 6 Mbps constant rate was used for transmitting ACK frames. Five different combinations have been simulated, i.e. the total number of stations (the number of nodes in group 1, the number of nodes in group 2) was 2(1,1), 5(3,2), 10(7,3), 15(11,4), and 20(14,6). We observed that the throughput when using

DCF and that of when using CRB are significantly decreased in the presence of hidden nodes[23]. In addition, the throughput when using CRB in the presence of hidden nodes is very close to that of using DCF in the presence of hidden nodes; however, the throughput when using CRB dose not get worse than that of using DCF. We also observed that the retransmission ratio per a data frame becomes very high (i.e. close to one) in the presence of hidden nodes.

D. Two overlapped APs

The two APs shown in Fig. 15(d) are assumed to be using the same frequency channel, and the number of connected STAs to each of the APs is ten. In this simulation, the log distance propagation loss model with Nakagami fading [24] was applied. A 54 Mbps constant rate was used for transmitting data frames, and a 6 Mbps constant rate was used for transmitting ACK frames. Fig. 18 is obtained from the average values of the last 0.2 second of each repeated simulation. We see that when the distance between the two APs ($= d$) varies from 5 meters to 500 meters, the throughput when using CRB is always higher than that of using DCF.

VI. DISCUSSION AND CONCLUSIONS

In this paper, we proposed the centralized random backoff (CRB) protocol and evaluated its performance. The evaluation results showed that when 1 second convergence time is allowed, 14 nodes using CRB can operate in a collision free state without changing the contention window size. Moreover, as time increases, the number of nodes that can operate in a collision free state increases (up to the maximum contention window size). However, in the deterministic backoff, the maximum number of nodes that can operate in a collision free state is limited to 8. Otherwise, it requires dynamic adjustment of the ring size when the number of nodes is varying. However, timely adjustment of the optimum ring size is a very complex issue, and it might be ineffective in practical wireless networks. Because of this CRB is more effective than SRB when the number of nodes varies with time. As future work, an *adaptive* VBA will be studied that resolves the lengthy convergence time issue when the value n is large. In addition, the fairness in the mixed network scenario should be improved, and a finite retransmission limit should be considered. We think CRB could be considered to enable efficient device to device (D2D) communications in licensed channels.

[23]A station can detect the presence of a hidden node by overhearing data frames and ACK frames, and it can inform the AP of the fact once in a while by transmitting a short and unique busytone signal just after receiving a specific beacon frame. We think that a unique subcarrier can be assigned to each station by the AP for this purpose in the connection establishment procedure. A similar use of unique subcarriers among stations (sharing a single AP) for notifying some information to the AP while not causing a collision can also be found in [26]. In this way, the AP can be made aware of the hidden node, and the AP can start to operate using CRB when no hidden node has been detected for a specific time period.

APPENDIX A
PROOF OF $D_i^l|_{(1st\ slot)}$

Assuming $m = 6$, the scalar $D_0^l|_{(1st\ slot)}$ (i.e. the probability of selecting a new non-zero SBC in Range 0 in the first successful slot time when the number of SBC is l) is obtained by equation (26),

$$\begin{aligned} D_0^l|_{(1st\ slot)} &= \left(\frac{W_0 - N_0^l - 1}{W_0}\right) + Q_0^l \left(\frac{W_0 - N_0^l - 1}{W_1}\right) + Q_0^l Q_1^l \left(\frac{W_0 - N_0^l - 1}{W_2}\right) \\ &+ Q_0^l Q_1^l Q_2^l \left(\frac{W_0 - N_0^l - 1}{W_3}\right) + Q_0^l Q_1^l Q_2^l Q_3^l \left(\frac{W_0 - N_0^l - 1}{W_4}\right) \\ &+ Q_0^l Q_1^l Q_2^l Q_3^l Q_4^l \left(\frac{W_0 - N_0^l - 1}{W_5}\right) + Q_0^l Q_1^l Q_2^l Q_3^l Q_4^l Q_5^l \left(\frac{W_0 - N_0^l - 1}{W_6}\right) \frac{1}{1 - Q_6^l} \end{aligned} \tag{26}$$

where the numerator $(W_0 - N_0^l - 1)$ represents available (i.e. non-allocated) non-zero numbers in Range 0. Equation (26) can be rewritten as relation (27) when $i = 0$.

$$D_i^l|_{(1st\ slot)} = \begin{cases} (W_0 - N_0^l - 1)\left(\dfrac{1}{W_0} + \displaystyle\sum_{y=0}^{m-2}\left(\frac{\prod_{k=0}^{y} Q_k^l}{W_{y+1}}\right) + \frac{\prod_{k=0}^{m-1} Q_k^l}{W_m(1 - Q_m^l)}\right) & i = 0 \\ (W_{i-1} - N_i^l)\left(\displaystyle\sum_{y=i-1}^{m-2}\left(\frac{\prod_{k=0}^{y} Q_k^l}{W_{y+1}}\right) + \frac{\prod_{k=0}^{m-1} Q_k^l}{W_m(1 - Q_m^l)}\right) & 1 \le i < m \\ (W_{m-1} - N_m^l)\dfrac{\prod_{k=0}^{m-1} Q_k^l}{W_m(1 - Q_m^l)} & i = m \end{cases} \tag{27}$$

Note that although the value of $D_i^l|_{(1st\ slot)}$ is expressed by the two variables N_i^l and Q_i^l in equation (27), considering equation (1) we find that the value of $D_i^l|_{(1st\ slot)}$ can be expressed by only N_i^l. Now, equations (4) and (27) can be used to obtain equation (5).

APPENDIX B
PROOF OF $b_{0,0}$

The state transition probabilities of the Markov Chain model depicted in Fig. 11 are given by the equation (9). Based on this, we obtain (28).

$$b_{i,k} = \frac{W_i - k}{W_i} \begin{cases} (1-p)P_i^l \displaystyle\sum_{j=0}^{m} b_{j,0} & i = 0 \\ p b_{i-1,0} + (1-p)P_i^l \displaystyle\sum_{j=0}^{m} b_{j,0} & 0 < i < m \\ p(b_{m-1,0} + b_{m,0}) + (1-p)P_i^l \displaystyle\sum_{j=0}^{m} b_{j,0} & i = m \end{cases} \tag{28}$$

If the value of l is zero in the equation (28), then $P_0^l = 1$ and $P_i^l = 0$ (given $i \in [1, m]$). In this case, the equation (28) becomes identical to that of legacy nodes presented in [2].

From the equations (10) and (28), we see equations in (29).

$$\begin{aligned} b_{i,0} &= pb_{i-1,0} + b_{0,0}\frac{P_i^l}{P_0^l} \;\rightarrow\; b_{i,0} = b_{0,0}\left(p^i + \sum_{j=0}^{i-1} p^{i-j-1}\frac{P_{j+1}^l}{P_0^l}\right) \qquad 0 < i < m \\ (1-p)b_{m,0} &= pb_{m-1,0} + b_{0,0}\frac{P_m^l}{P_0^l} \;\rightarrow\; b_{m,0} = \frac{b_{0,0}}{(1-p)}\left(p^m + \sum_{j=0}^{m-1} p^{m-j-1}\frac{P_{j+1}^l}{P_0^l}\right) \qquad i = m \end{aligned} \tag{29}$$

Using the equations in (29), we rewrite (28) as (30).

$$b_{i,k} = b_{0,0}\frac{W_i - k}{W_i}\begin{cases} 1 & i = 0 \\ p^i + \sum_{j=0}^{i-1}(p^{i-1-j})\frac{P_{j+1}^l}{P_0^l} & 1 \le i < m \\ \frac{p^m}{1-p} + \sum_{j=0}^{m-1}(p^{m-1-j})\frac{P_{j+1}^l}{(1-p)P_0^l} & i = m \end{cases} \tag{30}$$

The equation (30) shows that all the values of $b_{i,k}$ can be expressed as a function of $b_{0,0}$, p, and P_i^l. Now, by imposing the normalization condition (31) we obtain $b_{0,0}$ as (11),

$$\begin{aligned} 1 =& \sum_{i=0}^{m}\sum_{k=0}^{W_i-1} b_{i,k} = \sum_{k=0}^{W_0-1} b_{0,k} + \sum_{i=1}^{m-1}\sum_{k=0}^{W_i-1} b_{i,k} + \sum_{k=0}^{W_m-1} b_{m,k} \\ =& \; b_{0,0}\sum_{k=0}^{W_0-1}\frac{W_0-k}{W_0} + b_{0,0}\sum_{i=1}^{m-1}\sum_{k=0}^{W_i-1}\left[\frac{W_i-k}{W_i}\left(p^i + \sum_{j=0}^{i-1}\frac{(p^{i-1-j})P_{j+1}^l}{P_0^l}\right)\right] \\ &+ b_{0,0}\sum_{k=0}^{W_m-1}\frac{W_m-k}{W_m}\left(\frac{p^m}{1-p} + \sum_{j=0}^{m-1}\frac{(p^{m-1-j})P_{j+1}^l}{(1-p)P_0^l}\right) \\ =& \; b_{0,0}\frac{W_0+1}{2} + b_{0,0}\sum_{i=1}^{m-1}\left[\frac{W_i+1}{2}\left(p^i + \sum_{j=0}^{i-1} f_i^j(l)\right)\right] + b_{0,0}\frac{W_m+1}{2}\left(\frac{p^m}{1-p} + \sum_{j=0}^{m-1} f_m^j(l)\right) \end{aligned} \tag{31}$$

where $f_i^j(l) = (p^{i-1-j})P_{j+1}^l/P_0^l$ and $f_m^j(l) = (p^{m-1-j})P_{j+1}^l/P_0^l$ for short.

ACKNOWLEDGMENT

The authors would like to thank the editors (Prof. A. Banchs and Prof. G. Bianchi) and four anonymous reviewers for helpful comments that have improved the quality of the paper. The first author was supported by the Edinburgh Global Research Scholarship and the Principal's Career Development PhD Scholarship from the University of Edinburgh, Scotland, UK.

REFERENCES

[1] A. Gummalla and J. Limb, "Wireless medium access control protocols," *IEEE Communications Surveys and Tutorials*, vol. 9, pp. 1311–1321, Dec. 2000.

[2] G. Bianchi, "Performance analysis of the IEEE 802.11 distributed coordination function," *IEEE Journal on Selected Areas in Communications*, Dec. 2000.

[3] P. Chatzimisios, A. Boucouvalas, and V.Vitsas, "IEEE 802.11 packet delay a finite retry limit analysis," *IEEE GLOBECOM*, 2003.

[4] Y. Zheng, K. Lu, D. Wu, and Y. Fang, "Performance analysis of IEEE 802.11 DCF in imperfect channels," *IEEE Transactions on Vehicular Technology*, 2006.

[5] D. Malone, K. Duffy, and D. Leith, "Modeling the 802.11 distributed coordination function in nonsaturated heterogeneous conditions," *IEEE/ACM Transactions on Networking*, Nov. 2007.

[6] E. Felemban and E. Ekici, "Single hop IEEE 802.11 DCF analysis revisited: Accurate modeling of channel access delay and throughput for saturated and unsaturated traffic cases," *IEEE Transactions on Wireless Communications*, May 2011.

[7] G. Bianchi and I. Tinnirello, "Kalman filter estimation of the number of competing terminals in an IEEE 802.11 network," *IEEE INFOCOM*, 2003.

[8] A. Toledo, T. Vercauteren, and X. Wang, "Adaptive optimization of IEEE 802.11 DCF based on Bayesian estimation of the number of competing terminals," *IEEE Transactions on Mobile Computing*, 2006.

[9] F. Cali, M. Conti, and E. Gregori, "Dynamic tuning of the IEEE 802.11 protocol to achieve a theoretical throughput limit," *IEEE/ACM Transactions on Networking*, vol. 80, pp. 166–188, Dec. 2000.

[10] Z. Haas and J. Deng, "On optimizing the backoff interval for random access schemes," *IEEE Transactions on Communications*, vol. 51, no. 2, Feb. 2003.

[11] C. Wang, B. Li, and L. Li, "A new collision resolution mechanism to enhance the performance of IEEE 802.11 DCF," *IEEE Transactions on Vehicular Technology*, 2004.

[12] M. Heusse, F. Rousseau, R. Guillier, and A. Duda, "Idle sense: An optimal access method for high throughput and fairness in rate diverse wireless LANs," *ACM SIGCOMM Computer Communication Review*, 2005.

[13] D. Deng, C. Ke, H. Chen, and Y. Huang, "Contention window optimization for IEEE 802.11 DCF access control," *IEEE Transactions on Wireless Communications*, vol. 33, Jul. 2008.

[14] [Online]. Available: http://www.ieee802.org/11/Reports/tgax_update.htm

[15] J. Choi, J. Yoo, and S. Choi, "EBA: An enhancement of the IEEE 802.11 DCF via distributed reservation," *IEEE Transactions on Mobile Computing*, 2005.

[16] J. Lee and J. Walrand, "Zero collision random backoff algorithm," *IEEE INFOCOM*, 2007.

[17] Y. He, J. Sun, R. Yuan, and W. Gong, "A reservation based backoff method for video streaming in 802.11 home networks," *IEEE Journal on Selected Areas in Communications*, Jan. 2010.

[18] J. Barcelo, A. Toledo, C. Cano, and M. Oliver, "Fairness and convergence of CSMA with enhanced collision avoidance (ECA)," *IEEE International Conference on Communications (ICC)*, vol. 4, pp. 511–523, Dec. 2010.

[19] J. Barcelo, B. Bellata, C. Cano, A. Sfairopoulou, M. Oliver, and K. Verma, "Towards a collision-free WLAN: Dynamic parameter adjustment in CSMA/E2CA," *EURASIP Journal of Wireless Communications and Networking*, 2011.

[20] Y. He, J. Sun, X. Ma, A. Vasilakos, R. Yuan, and W. Gong, "Semi-random backoff: towards resource reservation for channel access in wireless LANs," *IEEE/ACM Transactions on Networking*, vol. 128, pp. 1610–1618, 2013.

[21] S. Misra and M. Khatua, "Semi-distributed backoff: Collision-aware migration from random to deterministic backoff," *IEEE Transactions on Mobile Computing*, Jan. 2015.

[22] M. Tuysuz and H. Mantar, "Exploiting the channel using uninterrupted collision free MAC adaptation over IEEE 802.11 WLANs," *Wireless Communications and Mobile Computing*, vol. 91, pp. 1227–1230, 2015.

[23] J. Kim, D. Laurenson, and J. Thompson, "Fair and efficient full duplex MAC protocol based on the IEEE 802.11 DCF," *IEEE PIMRC*, vol. 90, pp. 538–548, Mar. 2016.

[24] M. Stoffers and G. Riley, "Comparing the ns-3 propagation models," *IEEE International Symposium on Modeling, Analysis and Simulation of Computer and Telecommunication Systems*, 2012.

[25] M. Lacage and T. Henderson, "Yet another network simulator," *Proceeding from the 2006 workshop on NS-2: the IP network simulator, ACM*, 2006.

[26] S. Sen, R. Choudhury, and S. Nelakuditi, "No time to countdown: Migrating backoff to the frequency domain," *Proceedings of the 17th annual international conference on Mobile computing and networking. ACM*, 2011.